UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/992,823	11/14/2001	Daniel W. Wong	ATI.0100520	5879

34456 7590 08/21/2007

LARSON NEWMAN ABEL POLANSKY & WHITE, LLP
5914 WEST COURTYARD DRIVE
SUITE 200
AUSTIN, TX 78730

EXAMINER	
CHAI, LONGBIT	
ART UNIT	**PAPER NUMBER**
2131	

MAIL DATE	DELIVERY MODE
08/21/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	09/992,823	WONG ET AL.	
	Examiner	Art Unit	
	Longbit Chai	2131	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *14 August 2007*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-47,49-54,63 and 64* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-47,49-54,63 and 64* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *14 November 2001* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. Currently pending claims are 1 – 47, 49 – 54 and 63 – 64.

Response to Arguments

2. As per claim 1, Applicant asserts prior-art does not teach "sending a first encrypted routine of a software driver to a peripheral device". Examiner respectfully disagrees with the following reasons:

- Examiner notes the claim language "a first encrypted routine of a software driver" is interpreted as an encrypted routine (of either a encryption / decryption algorithm routine) that is used by the software driver. This interpretation is also consistent with (a) the disclosure of the instant application that indicates "a software driver is used to interface between one component of the system, such as a processor, and a peripheral component and the software driver incorporates sensitive data including encryption / decryption routines used for encrypting / decrypting data transmitted between the software driver and the peripheral device" (SPEC: Page 3 Line 10 – 15) and (b) is also consistent with its own claims 2 and claim 3 (see claim 2 & 3).

- Ciacelli teaches sending a encrypted version of a decryption algorithm routine to a peripheral device for the purpose of data decryption (Ciacelli: Column 2 Line 61 – 63 and Column 5 Line 43 – 45). However, Ciacelli does not disclose expressly that the decryption algorithm routine is used by a software driver of the peripheral device for data decryption.

- Crick teaches the decryption algorithm routine can be used by a software device driver for data decryption (Crick: Column 3 Line 50 – 51 and Figure 1 / Element 106 – 108: one of the component device driver within a layered software architecture has the capability of encrypting and decrypting the security sensitive data and as such the given device driver

requires / uses encryption / decryption algorithm routines in order to perform the encryption / decryption algorithm functions).

- Therefore, Ciacelli in view of Crick does teach sending a first encrypted routine of a software driver to a peripheral device, wherein the software driver is to interface with the peripheral device and as such Applicant's arguments are respectfully traversed.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> A person shall be entitled to a patent unless –
>
> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

3. Claims 1 – 3, 8 – 13, 16, 31 – 33, 38 – 43, 47, 49, 50 – 52, 54 and 63 – 64 are rejected under 35 U.S.C. 103(a) as being unpatentable over Ciacelli (U.S. Patent 6,236,727), and in view of Crick et al. (U.S. Patent 5,675,793).

As per claim 1, 31, 40, 47 and 49, Ciacelli teaches a method comprising the steps of:

sending a first encrypted routine of a software driver to a peripheral device, wherein the software driver is to interface with the peripheral device (Ciacelli: Column 2 Line 61 – 63 and Column 5 Line 43 – 45: Examiner notes Ciacelli teaches sending a encrypted version of a decryption algorithm routine to a peripheral device for the purpose of data decryption. Besides, Examiner notes the claim language "a first encrypted routine of a software driver" is interpreted as an encrypted routine (of either a encryption / decryption algorithm routine) that is used by the software driver. This interpretation is also consistent with (a) the

disclosure of the instant application that indicates "a software driver is used to interface between one component of the system, such as a processor, and a peripheral component and the software driver incorporates sensitive data including encryption / decryption routines used for encrypting / decrypting data transmitted between the software driver and the peripheral device" (SPEC: Page 3 Line 10 – 15) and (b) is also consistent with its own claims 2 and claim 3 (see claim 2 & 3)).

However, Ciacelli does not does not disclose expressly the decryption algorithm routine is used by a software driver for data decryption (i.e. interpreted as a first routine of a software driver that is consistent with the disclosure of the instant application that indicates "a software driver is used to interface between one component of the system, such as a processor, and a peripheral component and the software driver incorporates sensitive data including encryption / decryption routines used for encrypting / decrypting data transmitted between the software driver and the peripheral device" (SPEC: Page 3 Line 10 – 15)).

Crick teaches the decryption algorithm routine can be used by a software device driver for data decryption (Crick: Column 3 Line 50 – 51 and Figure 1 / Element 106 – 108: one of the component device driver within a layered software architecture has the capability of encrypting and decrypting the security sensitive data and as such the given device driver requires / uses encryption / decryption algorithm routines in order to perform the encryption / decryption algorithm functions).

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teaching of Crick within the system of Ciacelli because (a) Ciacelli teaches a hardware device performs an encryption and decryption functions (Ciacelli: Column 3 Line 50 – 52 and Column 5 Line 5 – 60) and (b) Crick teaches an encryption / decryption function can be included and separately performed in a component device driver

within a layered structure of software device drivers so that not only each layer of device drivers can be developed and tested independently one another (Crick: Column 3 Line 50 – 51 and Figure 1 / Element 106 – 108) but also reduces the overhead / burden of allocation (de-allocation) of memory working spaces by allocating, prior to executing software routines (e.g. component device drivers), an amount of memory necessary to satisfy the requirements of all the software routines (Crick: Column 4 Line 5 – 15).

decrypting, at the peripheral device, the first encrypted routine to generate a plaintext routine (Examiner notes the encrypted "encryption algorithm routine" must be first decrypted prior to the proper use by a software driver to perform the data decryption function); and

providing the plaintext routine to the software driver (Crick: Column 3 Line 50 – 51 and Figure 1 / Element 106 – 108: the decrypted "decryption algorithm routine" is then presented to the device driver for decrypting the security data).

As per claim 2, Ciacelli as modified teaches sending a encrypted version of a decryption algorithm routine to the device, instead of sending a encrypted version of a encryption algorithm routine to the device – i.e., Ciacelli as modified does not disclose expressly the first encrypted routine is an encrypted version of an encryption routine.

Examiner notes Ciacelli as modified teaches (a) Crick teaches the device driver can perform not only the decryption function but also the encryption function and thus the encryption algorithm routine should be needed by a device driver as well (Crick: Column 3 Line 50 – 51) and (b) Ciacelli teaches first, sending the encryption version of the decryption algorithm to the hardware device (Ciacelli: Column 5 Line 43 – 45) and further teaches re-encrypting security data when forwarding the data down to a chain of multiple devices (Ciacelli: Column 4 Line 42 –

45, Column 5 Line 54 – 60 and Column 7 Line 58 – 65). Therefore, It would have been obvious to a person of ordinary skill in the art at the time the invention was made to recognize the encryption algorithm routine is equally needed on downloading a pair of a decryption / encryption algorithm routines in order to perform the re-encryption function after decrypting the data.

As per claim 3 and 50, Ciacelli as modified teaches the first encrypted routine is an encrypted version of a decryption routine (See the same rationale of rejection as set forth above in claim 1 of this Section).

As per claim 8, 33 and 38, Ciacelli as modified teaches sending a decryption code to the peripheral device, where the decryption code is to be used by the peripheral device to decrypt the first encrypted routine (Ciacelli: Column 5 Line 45 – 60).

As per claim 9, Ciacelli as modified teaches removing the plaintext routine (Ciacelli: Column 7 Line 16 – 21).

As per claim 10 and 54, Ciacelli as modified teaches encrypting, at the peripheral device, the plaintext routine to generate a second encrypted routine, where the second encrypted routine is a version of the first encrypted routine (Examiner notes Ciacelli teaches first, sending the encryption version of the decryption algorithm to the hardware device (Ciacelli: Column 5 Line 43 – 45) and further teaches re-encrypting security data when forwarding the data down to a chain of multiple devices (Ciacelli: Column 4 Line 42 – 45, Column 5 Line 54 – 60 and Column 7 Line 58 – 65). Therefore, It would have been obvious to a person of ordinary skill in the art at the time the invention was made to recognize the downloading of decryption /

encryption algorithm routines are equally needed by each device on a chain of multiple devices in order to perform the re-encryption function after decrypting the data – also see the same rationale addressed in claim 17).

As per claim 11, Ciacelli as modified teaches sending a encryption code to the peripheral device, where the encryption code is to be used by the peripheral device to encrypt the plaintext routine (Ciacelli: Column 6 Line 42 – 45: an encryption key can be also qualified as an encryption code – i.e. key code).

As per claim 12 and 32, Ciacelli as modified teaches the second encrypted routine is a modified version of the first encrypted routine (Ciacelli: Column 7 Line 24 – 29 and Column 5 Line 55 – 56: the encrypted "decryption / encryption algorithm routines" can be updated changed on a needed basis).

As per claim 13, Ciacelli as modified teaches selecting the first encrypted routine from a plurality of different encrypted routines, wherein the plurality of different encrypted routines are functionally equivalent (Ciacelli: Column 5 Line 55 – 57).

As per claim 16, Ciacelli as modified teaches providing includes storing the plaintext routine in a location in memory accessible by the software driver, and where the location in memory is known to the software driver (Crick: Column 3 Line 50 – 51 and Figure 1 / Element 106 – 108: the decrypted "encryption / decryption algorithm routine" is then presented to the device driver for encrypting / decrypting the security data).

As per claim 39, Ciacelli as modified teaches sending a encryption code to the peripheral device, where the encryption code is to be used by the peripheral device to encrypt

the plaintext routine (Ciacelli: Column 6 Line 42 – 45: an encryption key is qualified as an encryption code – i.e. key code).

As per claim 41, Ciacelli as modified teaches said first interface and said second interface are implemented using a same interface (Ciacelli: see for example: Column 5 Line 43 – 48: the same interface of decryption module to receive and execute the decryption function for encrypted routine).

As per claim 42, the claim limitations are met as the same reasons as that set forth above in rejecting claim 10 and 11.

As per claim 43, Ciacelli as modified teaches the first hardware component and the second component are implemented using a same hardware component (Ciacelli: see for example: Column 5 Line 43 – 48: the same hardware component of decryption module to receive and execute the decryption function for encrypted routine).

As per claim 51, Ciacelli as modified teaches the first encrypted data includes an encrypted version of one of: a private encryption key, a private decryption key, a chip ID, and a device ID (Ciacelli: Column 6 Line 42 – 45).

As per claim 52, Ciacelli as modified teaches the application includes a software driver (Crick: Column 3 Line 50 – 51 and Figure 1 / Element 106 – 108: one of the component device driver within a layered software architecture has the capability of encrypting and decrypting the security sensitive data and as such the given device driver requires / uses encryption /

decryption algorithm routines in order to perform the encryption / decryption algorithm functions).

As per claim 63, Ciacelli as modified teaches processing data at the peripheral device using the plain text routine (Crick: Column 5 Line 45 – 60: the unencrypted routine (e.g. after the decryption) is considered as the plain text routine that can be used to proceed the encryption or decryption functions for the sensitive data).

As per claim 64, Ciacelli as modified teaches decrypting data at the peripheral device using the plain text routine (Crick: Column 5 Line 45 – 60: the unencrypted routine (e.g. after the decryption) is considered as the plain text routine that can be used to proceed the encryption or decryption functions for the sensitive data).

4. Claims 4, 5, 7, 17 – 19, 21 – 27, 30, 34, 35, 37, 44, 46 and 53 are rejected under 35 U.S.C. 103(a) as being unpatentable over Ciacelli (U.S. Patent 6,236,727), and in view of Crick et al. (U.S. Patent 5,675,793), in view of Freeman (U.S. Patent 2002/0129374).

As per claim 17, Ciacelli teaches a method comprising:

sending a first encrypted routine of a software driver to a hardware device, wherein the software driver is to interface with the hardware device, and where the first encrypted routine is an encrypted version of a encryption routine (Ciacelli: Column 2 Line 61 – 63 and Column 5 Line 43 – 45: *Examiner notes*: Ciacelli teaches sending a encrypted version of a decryption algorithm routine to a peripheral device for the purpose of data

decryption). However, Ciacelli does not teach (a) the decryption algorithm routine is used by a software driver for data decryption (i.e. interpreted as a first routine of a software driver that is consistent with the disclosure of the instant application (SPEC: Page 3 Line 10 – 15: also see Response to Argument)), and (b) Ciacelli teaches sending a encrypted version of a decryption algorithm routine to the device for data decryption, instead of sending a encrypted version of a encryption algorithm routine to the device for data encryption. The rationale of rejections for (a) and (b) are explored in detail as follow.

- ✓ For part (a): **Ciacelli does not teach "the decryption algorithm routine is used by a software driver for data decryption".**

  Crick teaches the decryption algorithm routine can be used by a software device driver (Crick: Column 3 Line 50 – 51 and Figure 1 / Element 106 – 108: one of the component device driver within a layered software architecture has the capability of encrypting and decrypting the security sensitive data and as such the given device driver requires / uses encryption / decryption algorithm routines in order to perform the encryption / decryption algorithm functions).

  It would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teaching of Crick within the system of Ciacelli because (a) Ciacelli teaches a hardware device performs an encryption and decryption functions (Ciacelli: Column 3 Line 50 – 52 and Column 5 Line 5 – 60) and (b) Crick teaches an encryption / decryption function can be included and separately performed in a component device driver within a layered structure of software device drivers so that not only each layer of device drivers can be developed and tested independently one another (Crick: Column 3 Line 50 – 51 and Figure 1 / Element 106 – 108) but also reduces the overhead /

burden of allocation (de-allocation) of memory working spaces by allocating, prior to executing software routines (e.g. component device drivers), an amount of memory necessary to satisfy the requirements of all the software routines (Crick: Column 4 Line 5 – 15).

✓ For part (b): **Ciacelli does not teach "sending a encrypted version of a encryption algorithm routine to the device for data encryption",** even though Ciacelli does teache "sending a encrypted version of a decryption algorithm routine to the device for data encryption".

Examiner notes (a) Crick teaches the device driver can perform not only the decryption function but also the encryption function and thus the encryption algorithm routine should be needed by a device driver as well (Crick: Column 3 Line 50 – 51) and (b) Ciacelli teaches first, sending the encryption version of the decryption algorithm to the hardware device (Ciacelli: Column 5 Line 43 – 45) and further teaches re-encrypting security data when forwarding the data down to a chain of multiple devices (Ciacelli: Column 4 Line 42 – 45, Column 5 Line 54 – 60 and Column 7 Line 58 – 65). Therefore, It would have been obvious to a person of ordinary skill in the art at the time the invention was made to recognize the encryption algorithm routine is equally needed on downloading a pair of a decryption / encryption algorithm routines in order to perform the re-encryption function after decrypting the data.

Ciacelli does not teach the **hardware device is a graphic chip**.

Freeman teaches the hardware device is a graphic chip (Freeman: see for example, Paragraph [0117]).

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teaching of Freeman within the system of Ciacelli because (a) Ciacelli discloses the video multimedia content scrambling system (CSS) and Moving Picture Expert Group (MPEG) standard (Ciacelli: Column 3 Line 25 – 43 and Column 2 Line 48 – 50 and (b) Freeman teaches using a graphic chip to realize the MPEG adaptation and to process the video data stream (Freeman: Paragraph [0117] and Figure 7 Element 376 & 388).

decrypting, at the graphics chip, the first encrypted routine to generate a plaintext routine, wherein the plaintext routine is a version of the encryption routine (Examiner notes the encrypted "encryption algorithm routine" must be first decrypted prior to the proper use by a software driver to perform the data encryption function); and

storing the plaintext routine in memory in a location known to the software driver (Crick: Column 3 Line 50 – 51 and Figure 1 / Element 106 – 108: the decrypted "encryption algorithm routine" is then presented to the device driver for encrypting the security data).

As per claim 4, 34 and 53, Ciacelli does not disclose expressly the peripheral device is a graphics chip.

Freeman teaches the hardware device is a graphic chip (Freeman: see for example, Paragraph [0117]).

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teaching of Freeman within the system of Ciacelli because (a) Ciacelli discloses the video multimedia content scrambling system (CSS) and Moving Picture Expert Group (MPEG) standard (Ciacelli: Column 3 Line 25 – 43 and Column 2 Line 48 – 50 and (b) Freeman teaches using a graphic chip to realize the MPEG adaptation and to process the video data stream (Freeman: Paragraph [0117] and Figure 7 Element 376 & 388).

As per claim 5, 35 and 44, Ciacelli as modified teaches decrypting is performed by a graphics chip (Ciacelli: see for example: Column 3 Line 25 – 43, Column 5 Line 43 – 60 and Column 2 Line 48 – 50) & (Freeman: see for example, Paragraph [0117]).

Ciacelli in view of Freeman does not disclose expressly decrypting is performed by a 3D pipe of the graphics chip.

However, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to modify Ciacelli to accommodate decrypting is performed by a 3D pipe of the graphics chip because Ciacelli in view of Freeman teaches decrypting / encrypting can be performed by multiple peripheral devices in MPEG video data encryption techniques (Ciacelli: see for example, Column 7 Line 58 – 65 and Column 2 Line 48 – 50) and 3D (3-Dimension) engine is merely one part of a series of video graphic chips in this claimed subject of matter to perform encryption / decryption).

As per claim 7, 37 and 46, Ciacelli as modified teaches decrypting is performed by dedicated encryption hardware of the graphics chip (Ciacelli: see for example: Abstract Line 15 – 17 and Column 2 Line 55 – 63) & (Freeman: see for example, Paragraph [0117]).

As per claim 18, Ciacelli as modified teaches sending a decryption code to the peripheral device, where the decryption code is to be used by the peripheral device to decrypt the first encrypted routine (Ciacelli: Column 5 Line 45 – 60).

As per claim 19, Ciacelli as modified teaches decrypting is performed by a graphics chip (Ciacelli: Column 3 Line 25 – 43, Column 5 Line 43 – 60 and Column 2 Line 48 – 50) & (Freeman: Paragraph [0117]).

Ciacelli as modified does not disclose expressly decrypting is performed by a 3D pipe of the graphics chip.

However, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to modify Ciacelli to accommodate decrypting is performed by a 3D pipe of the graphics chip because Ciacelli in view of Freeman teaches decrypting / encrypting can be performed by multiple peripheral devices in MPEG video data encryption techniques (Ciacelli: see for example, Column 7 Line 58 – 65 and Column 2 Line 48 – 50) and 3D (3-Dimension) engine is merely one part of a series of video graphic chips in this claimed subject of matter to perform encryption / decryption).

As per claim 21, Ciacelli as modified teaches decrypting is performed by dedicated encryption hardware of the graphics chip (Ciacelli: see for example: Abstract Line 15 – 17 and Column 2 Line 55 – 63) & (Freeman: see for example, Paragraph [0117]).

As per claim 22, Ciacelli as modified teaches decrypting is performed through a series of components coupled within the graphics chip (Ciacelli: see for example: Column 7 Line 58 – 65) & (Freeman: see for example, Paragraph [0117]).

As per claim 23, Ciacelli as modified teaches removing the plaintext routine (Ciacelli: see for example: Column 7 Line 16 – 21).

As per claim 24, Ciacelli as modified teaches encrypting, at the peripheral device, the plaintext routine to generate a second encrypted routine, where the second encrypted routine is a version of the first encrypted routine (Examiner notes Ciacelli teaches first, sending the

encryption version of the decryption algorithm to the hardware device (Ciacelli: Column 5 Line 43 – 45) and further teaches re-encrypting security data when forwarding the data down to a chain of multiple devices (Ciacelli: Column 4 Line 42 – 45, Column 5 Line 54 – 60 and Column 7 Line 58 – 65). Therefore, It would have been obvious to a person of ordinary skill in the art at the time the invention was made to recognize the downloading of decryption / encryption algorithm routines are equally needed by each device on a chain of multiple devices in order to perform the re-encryption function after decrypting the data – also see the same rationale addressed in claim 17).

As per claim 25, Ciacelli as modified teaches sending a encryption code to the peripheral device, where the encryption code is to be used by the peripheral device to encrypt the plaintext routine (Ciacelli: Column 6 Line 42 – 45: an encryption key can be also qualified as an encryption code – i.e. key code. Regarding encrypting the plaintext routine, see the same rationale addressed above in claim 10).

As per claim 26, Ciacelli as modified teaches the second encrypted routine is a modified version of the first encrypted routine (Ciacelli: Column 7 Line 24 – 29 and Column 5 Line 55 – 56: the encrypted "decryption / encryption algorithm routines" can be updated changed on a needed basis).

As per claim 27, Ciacelli as modified teaches selecting the first encrypted routine from a plurality of different encrypted routines, wherein the plurality of different encrypted routines are functionally equivalent (Ciacelli: see for example: Column 14 Line 10 – 15).

As per claim 30, Ciacelli as modified teaches providing includes storing the plaintext routine in a location in memory accessible by the software driver, and where the location in memory is known to the software driver (Crick: Column 3 Line 50 – 51 and Figure 1 / Element 106 – 108: the decrypted "encryption / decryption algorithm routine" is then presented to the device driver for encrypting / decrypting the security data).

5. Claims 6, 20, 36 and 45 are rejected under 35 U.S.C. 103(a) as being unpatentable over Ciacelli (U.S. Patent 6,236,727), and in view of Crick et al. (U.S. Patent 5,675,793), in view of Freeman (U.S. Patent 2002/0129374), and in view of Ho (U.S. Patent 5,495,432).

As per claim 6, 36 and 45, Ciacelli as modified teaches decrypting is performed by a graphics chip (Ciacelli: Column 3 Line 25 – 43, Column 5 Line 43 – 60 and Column 2 Line 48 – 50) & (Freeman: Paragraph [0117]). Ciacelli as modified does not disclose expressly decrypting is performed by a IDCT component of the graphics chip.

Ho teaches decrypting is performed by a IDCT component of the graphics chip (Ho: Column 13 Line 35 – 37).

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to to combine the teaching of Ho within the system of Ciacelli as modified because (a) Ciacelli discloses the video multimedia content scrambling system (CSS) and Moving Picture Expert Group (MPEG) standard (Ciacelli: Column 3 Line 25 – 43 and Column 2 Line 48 – 50 and (b) Ho teaches using a IDCT component of a graphic functionality to realize the MPEG adaptation and to process the video data stream (Ho: Column 13 Line 35 – 37).

As per claim 20, Ciacelli as modified teaches decrypting is performed by a graphics chip (Ciacelli: Column 3 Line 25 – 43, Column 5 Line 43 – 60 and Column 2 Line 48 – 50) &

(Freeman: Paragraph [0117]). Ciacelli as modified does not disclose expressly decrypting is performed by a IDCT component of the graphics chip.

Ho teaches decrypting is performed by a IDCT component of the graphics chip (Ho: Column 13 Line 35 – 37).

See the same rationale of combination in rejecting claim 6.

6. Claims 14 and 15 are rejected under 35 U.S.C. 103(a) as being unpatentable over Ciacelli (U.S. Patent 6,236,727), and in view of Crick et al. (U.S. Patent 5,675,793), and in view of Wilson (U.S. Patent 4,520,232).

As per claim 14, Ciacelli does not disclose expressly decrypting includes using a map as a decryption key.

Wilson teaches decrypting includes using a map as a decryption key (Wilson: see for example: Column 2 Line 12 – 24).

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teaching of Wilson within the system of Ciacelli because (a) Ciacelli discloses the video multimedia content scrambling system (CSS) and Moving Picture Expert Group (MPEG) standard (Ciacelli: Column 3 Line 25 – 43 and Column 2 Line 48 – 50 and (b) Wilson teaches providing a poly-graphic encryption mechanism which is both fast and inexpensive with enhanced security strength (Wilson: see for example, Column 1 Line 28 – 34).

As per claim 15, Ciacelli as modified teaches the map includes a texture map (Wilson: see for example, Column 1 Line 28 – 34: the matrix is qualified as a two-dimensional texture map).

7. Claims 28 and 29 are rejected under 35 U.S.C. 103(a) as being unpatentable over Ciacelli (U.S. Patent 6,236,727), and in view of Crick et al. (U.S. Patent 5,675,793), in view of Freeman (U.S. Patent 2002/0129374), and in view of Wilson (U.S. Patent 4,520,232).

As per claim 28, Ciacelli as modified does not disclose expressly decrypting includes using a map as a decryption key.

Wilson teaches decrypting includes using a map as a decryption key (Wilson: see for example: Column 2 Line 12 – 24).

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teaching of Wilson within the system of Ciacelli as modified because (a) Ciacelli discloses the video multimedia content scrambling system (CSS) and Moving Picture Expert Group (MPEG) standard (Ciacelli: Column 3 Line 25 – 43 and Column 2 Line 48 – 50 and (b) Wilson teaches providing a poly-graphic encryption mechanism which is both fast and inexpensive with enhanced security strength (Wilson: see for example, Column 1 Line 28 – 34).

As per claim 29, Ciacelli as modified teaches teaches the map includes a texture map (Wilson: see for example, Column 1 Line 28 – 34: the matrix is qualified as a two-dimensional texture map).

Conclusion

THIS ACTION IS MADE FINAL. Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Longbit Chai whose telephone number is 571-272-3788. The examiner can normally be reached on Monday-Friday 9:00am-5:00pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Ayaz R. Sheikh can be reached on 571-272-3795. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Longbit Chai
Examiner
Art Unit 2131

LBC

SYED A. ZIA
PRIMARY EXAMINER